Contrastive Attention Mechanism for Abstractive Sentence Summarization

Xiangyu Duan[1,2]*, Hongfei Yu[2]*, Mingming Yin[2], Min Zhang[1,2], Weihua Luo[3], Yue Zhang[4]

[1] *Institute of Artificial Intelligence, Soochow University, Suzhou, China*
[2] *School of Computer Science and Technology, Soochow University, Suzhou, China*
[3] *Alibaba DAMO Academy, Hangzhou, China*
[4] *School of Engineering, Westlake University, China*

xiangyuduan@suda.edu.cn; {hfyu,mmyin}@stu.suda.edu.cn; minzhang@suda.edu.cn
weihua.luowh@alibaba-inc.com; yue.zhang@wias.org.cn

Abstract

We propose a contrastive attention mechanism to extend the sequence-to-sequence framework for abstractive sentence summarization task, which aims to generate a brief summary of a given source sentence. The proposed contrastive attention mechanism accommodates two categories of attention: one is the conventional attention that attends to relevant parts of the source sentence, the other is the opponent attention that attends to irrelevant or less relevant parts of the source sentence. Both attentions are trained in an opposite way so that the contribution from the conventional attention is encouraged and the contribution from the opponent attention is discouraged through a novel softmax and softmin functionality. Experiments on benchmark datasets show that, the proposed contrastive attention mechanism is more focused on the relevant parts for the summary than the conventional attention mechanism, and greatly advances the state-of-the-art performance on the abstractive sentence summarization task. We release the code at `https://github.com/travel-go/Abstractive-Text-Summarization`.

1 Introduction

Abstractive sentence summarization aims at generating concise and informative summaries based on the core meaning of source sentences. Previous endeavors tackle the problem through either rule-based methods (Dorr et al., 2003) or statistical models trained on relatively small scale training corpora (Banko et al., 2000). Following its successful applications on machine translation (Sutskever et al., 2014; Bahdanau et al., 2015), the sequence-to-sequence framework is also applied on the abstractive sentence summarization task using large-scale sentence summary corpora (Rush et al., 2015; Chopra et al., 2016; Nallapati et al., 2016), obtaining better performance compared to the traditional methods.

One central component in state-of-the-art sequence to sequence models is the use of attention for building connections between the source sequence and target words, so that a more informed decision can be made for generating a target word by considering the most relevant parts of the source sequence (Bahdanau et al., 2015; Vaswani et al., 2017). For abstractive sentence summarization, such attention mechanisms can be useful for selecting the most salient words for a short summary, while filtering the negative influence of redundant parts.

We consider improving abstractive summarization quality by enhancing target-to-source attention. In particular, a contrastive mechanism is taken, by encouraging the contribution from the conventional attention that attends to relevant parts of the source sentence, while at the same time penalizing the contribution from an opponent attention that attends to irrelevant or less relevant parts. Contrastive attention was first proposed in computer vision (Song et al., 2018a), which is used for person re-identification by attending to person and background regions contrastively. To our knowledge, we are the first to use contrastive attention for NLP and deploy it in the sequence-to-sequence framework.

In particular, we take Transformer (Vaswani et al., 2017) as the baseline summarization model, and enhance it with a proponent attention module and an opponent attention module. The former acts as the conventional attention mechanism, while the latter can be regarded as a dual module to the former, with similar weight calculation structure, but using a novel softmin function to discourage contributions from irrelevant or less relevant words.

To our knowledge, we are the first to investigate

[*] Equal contribution.

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 3035–3044,
Hong Kong, China, November 3–7, 2019. ©2019 Association for Computational Linguistics

Transformer as a sequence to sequence summarizer. Results on three benchmark datasets show that it gives highly competitive accuracies compared with RNN and CNN alternatives. When equipped with the proposed contrastive attention mechanism, our Transformer model achieves the best reported results on all data. The visualization of attentions shows that through using the contrastive attention mechanism, our attention is more focused on relevant parts than the baseline. We release our code at XXX.

2 Related Work

Automatic summarization has been investigated in two main paradigms: the extractive method and the abstractive method. The former extracts important pieces of source document and concatenates them sequentially (Jing and McKeown, 2000; Knight and Marcu, 2000; Neto et al., 2002), while the latter grasps the core meaning of the source text and re-state it in short text as abstractive summary (Banko et al., 2000; Rush et al., 2015). In this paper, we focus on abstractive summarization, and especially on abstractive sentence summarization.

Previous work deals with the abstractive sentence summarization task by using either rule based methods (Dorr et al., 2003), or statistical methods utilizing a source-summary parallel corpus to train a machine translation model (Banko et al., 2000), or a syntax based transduction model (Cohn and Lapata, 2008; Woodsend et al., 2010).

In recent years, sequence-to-sequence neural framework becomes predominant on this task by encoding long source texts and decoding into short summaries together with the attention mechanism. RNN is the most commonly adopted and extensively explored architecture (Chopra et al., 2016; Nallapati et al., 2016; Li et al., 2017). A CNN-based architecture is recently employed by Gehring *et al.* (2017) using ConvS2S, which applies CNN on both encoder and decoder. Later, Wang *et al.* (2018) build upon ConvS2S with topic words embedding and encoding, and train the system with reinforcement learning.

The most related work to our contrastive attention mechanism is in the field of computer vision. Song *et al.* (2018a) first propose the contrastive attention mechanism for person re-identification. In their work, based on a pre-provided person and background segmentation, the two regions are contrastively attended so that they can be easily discriminated. In comparison, we apply the contrastive attention mechanism for sentence level summarization by contrastively attending to relevant parts and irrelevant or less relevant parts. Furthermore, we propose a novel softmax softmin functionality to train the attention mechanism, which is different to Song *et al.* (2018a), who use mean squared error loss for attention training.

Other explorations with respect to the characteristics of the abstractive summarization task include copying mechanism that copies words from source sequences for composing summaries (Gu et al., 2016; Gulcehre et al., 2016; Song et al., 2018b), the selection mechanism that elaborately selects important parts of source sentences (Zhou et al., 2017; Lin et al., 2018), the distraction mechanism that avoids repeated attention on the same area (Chen et al., 2016), and the sequence level training that avoids exposure bias in teacher forcing methods (Ayana et al., 2016; Li et al., 2018; Edunov et al., 2018). Such methods are built on conventional attention, and are orthogonal to our proposed contrastive attention mechanism.

3 Approach

We use two categories of attention for summary generation. One is the conventional attention that attends to relevant parts of source sentence, the other is the opponent attention that contrarily attends to irrelevant or less relevant parts. Both categories of attention output probability distributions over summary words, which are jointly optimized by encouraging the contribution from the conventional attention and discouraging the contribution from the opponent attention.

Figure 1 illustrates the overall networks. We use Transformer architecture as our basis, upon which we build the contrastive attention mechanism. The left part is the original Transformer. We derive the opponent attention from the conventional attention which is the encoder-decoder attention of the original Transformer, and stack several layers on top of the opponent attention as shown in the right part of Figure 1. Both parts contribute to the summary generation by producing probability distributions over the target vocabulary, respectively. The left part outputs **the conventional probability** based on the conventional attention as the original Transformer does, while the right part outputs **the opponent probability** based on the opponent

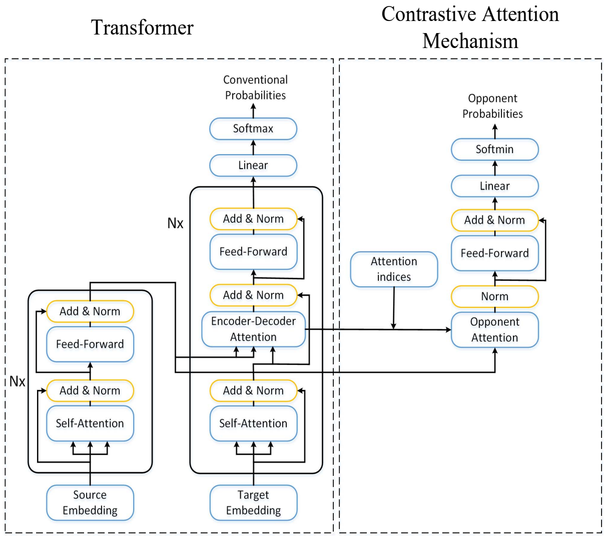

Figure 1: Overall networks. The left part is the original Transformer. The right part that takes the opponent attention as bottom layer fulfils the contrastive attention mechanism.

attention. The two probabilities in Figure 1 are jointly optimized in a novel way as explained in Section 3.3.

3.1 Transformer for Abstractive Sentence Summarization

Transformer is an attention network based sequence-to-sequence architecture (Vaswani et al., 2017), which encodes the source text into hidden vectors and decodes into the target text based on the source side information and the target generation history. In comparison to the RNN based architecture and the CNN based architecture, both the encoder and the decoder of Transformer adopt attention as main function.

Let X and Y denote the source sentence and its summary, respectively. Transformer is trained to maximize the probability of Y given X: $\prod_i P_c(y_i|y_1^{i-1}, X)$, where $P_c(y_i|y_1^{i-1}, X)$ is the conventional probability of the current summary word y_i given the source sentence and the summary generation history. P_c is computed based on the attention mechanism and the stacked deep layers as shown in the left part of Figure 1.

Attention Mechanism

Scaled dot-product attention is applied in Transformer:

$$\mathrm{attention}(Q, K, V) = \mathrm{softmax}(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}})V \quad (1)$$

where Q, K, V denotes query vector, key vectors, and value vectors, respectively. d_k denotes the dimension of one vector of K. Softmax function outputs the attention weights distributed over V. $\mathrm{attention}(Q, K, V)$ is a vector of weighted sum of elements of V, and represents current context information.

We focus on the encoder-decoder attention, which builds the connection between source and target by informing the decoder which area of the source text should be attended to. Specifically, in the encoder-decoder attention, Q is the single vector coming from the current position of the decoder, K and V are the same sequence of vectors that are the outcomes of the encoder at all source positions. Softmax function distributes the **attention weights** over the source positions.

The attentions in Transformer adopts the multi-head implementation, in which each head computes attention as Equation (1) but with smaller Q, K, V whose dimension is $1/h$ times of their original dimension respectively. The attentions from h heads are concatenated together and linearly projected to compose the final attention. In this way, multi-head attention provides a multi-view of attention behavior beneficial for the final performance.

Deep Layers

The "N×" plates in Figure 1 stands for the stacked N identical layers. On the source side, each layer of the stacked N layers contains two sublayers: the self-attention mechanism, and the fully connected feed-forward network. Each sublayer employs residual connection that adds input to outcome of sublayer, then layer normalization is employed on the outcome of the residual connection.

On the target summary side, each layer contains an additional sublayer of the encoder-decoder attention between the self-attention sublayer and the feed-forward sublayer. At the top of the decoder, the softmax layer is applied to convert the decoder output to summary word generation probabilities.

3.2 Contrastive Attention Mechanism

3.2.1 Opponent Attention

As illustrated in Figure 1, the opponent attention is derived from the conventional encoder-decoder attention. Since the multi-head attention is employed in Transformer, there are N×h heads in total in the conventional encoder-decoder attention, where N denotes the number of layers, h denotes

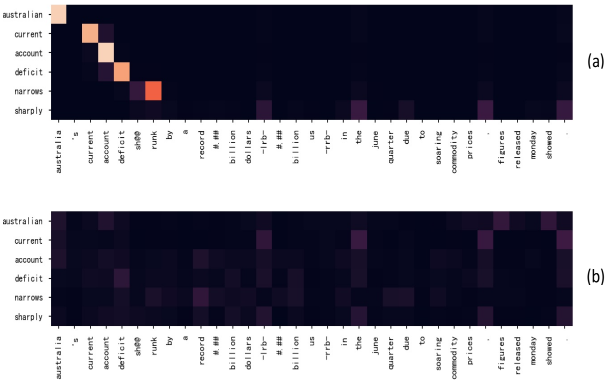

Figure 2: Heatmaps of two sampled heads from the conventional encoder-decoder attention. (a) is of the fifth head of the third layer, and (b) is of the fifth head of the first layer.

the number of heads in each layer. These heads exhibit diverse attention behaviors, posing a challenge of determining which head to derive the opponent attention, so that it attends to irrelevant or less relevant parts.

Figure 2 illustrates the attention weights of two sampled heads. The attention weights in (a) well reflect the word level relevant relation between the source sentence and the target summary, while attention weights in (b) do not. We find that such behavior characteristic of each head is fixed. For example, head (a) always exhibits the relevant relation across different sentences and different runs. Based on depicting heatmaps of all heads for a few sentences, we choose the head that corresponds well to the relevant relation between source and target to derive the opponent attention [1].

Specifically, let α_c denote the conventional encoder-decoder attention weights of the head which is used for deriving the opponent attention:

$$\alpha_c = \text{softmax}(\frac{qk^{\text{T}}}{\sqrt{d_k}}) \tag{2}$$

where q and k are from the head same to that of α_c. Let α_o denote the opponent attention weights. It is obtained through the opponent function applied on α_c followed by the softmax function:

$$\alpha_o = \text{softmax}(\text{opponent}(\alpha_c)) \tag{3}$$

The opponent function in Equation (3) performs a masking operation, which finds the maximum weight in α_c, and replaces it with the negative infinity value, so that the softmax function outputs zero given the negative infinity value input. Then the maximum weight in α_c is set zero in α_o after the opponent and softmax functions. In this way, the most relevant part of the source sequence, which receives maximum attention in the conventional attention weights α_c, is masked and neglected in α_o. Instead, the remaining less relevant or irrelevant parts are extracted into α_o for the following contrastive training and decoding.

We also tried other methods to calculate the opponent attention weights, such as $\alpha_o = \text{softmax}(1 - \alpha_c)$ (Song et al., 2018a) [2] or $\alpha_o = \text{softmax}(1/\alpha_c)$, which aims to make α_o contrary to α_c, but they underperform the masking opponent function on all benchmark datasets. So we present only the masking opponent in the following sections.

After α_o is obtained via Equation (3), the opponent attention is: $\text{attention}_o = \alpha_o v$, where v is from the head same to that of q and k in computing α_c.

Compared to the conventional attention attention_c, which summarizes current relevant context, attention_o summarizes current irrelevant or less relevant context. They constitute a contrastive pair, and contribute together for the final summary word generation.

3.2.2 Opponent Probability

The opponent probability $P_o(y_i|y_1^{i-1}, X)$ is computed by stacking several layers on top of attention_o, and a softmin layer in the end as shown in the right part of Figure (1). In particular,

$$\begin{aligned} z_1 &= \text{LayerNorm}(\text{attention}_o) && (4)\\ z_2 &= \text{FeedForward}(z_1) && (5)\\ z_3 &= \text{LayerNorm}(z_1 + z_2) && (6)\\ P_o(y_i|y_1^{i-1}, X) &= \text{softmin}(Wz_3) && (7) \end{aligned}$$

where W is the matrix of the linear projection sublayer.

attention_o contributes to P_o via Equation (4-7) step by step. The LayerNorm and FeedForward layers with residual connection is similar to the

[1] Given manual alignments between source and target of sampled sentence-summary pairs, we select the head that has the lowest alignment error rate (AER) of its attention weights.

[2] Song *et al.* (2018a) directly let $\alpha_o = 1 - \alpha_c$ in extracting background features for person re-identification in computer vision. We have to add softmax function since the attention weights must be normalized to one in sequence-to-sequence framework.

original Transformer, while a novel softmin function is introduced in the end to invert the contribution from $\mathrm{attention}_o$:

$$\mathrm{softmin}(v_i) = \frac{e^{(-v_i)}}{\sum^j e^{(-v_j)}} \tag{8}$$

where $v = Wz_3$, i.e., the input vector to the softmin function in Equation (7). Softmin normalizes v so that scores of all words in the summary vocabulary sum to one. We can see that the bigger the v_i, the smaller the $P_{o,i}$ is.

Softmin functions contrarily to softmax. As a result, when we try to maximize $P_o(y_i = y|y_1^{i-1}, X)$, where y is the gold summary word, we effectively search for an appropriate $\mathrm{attention}_o$ to generate the lowest v_g, where g is the index of y in v. It means that the more irrelevant is $\mathrm{attention}_o$ to the summary, the lower the v_g can be obtained, resulting in higher P_o.

3.3 Training and Decoding

During training, we jointly maximize the conventional probability P_c and the opponent probability P_o:

$$\mathcal{J} = \log(P_c(y_i|y_1^{i-1}, X) + \lambda \log(P_o(y_i|y_1^{i-1}, X) \tag{9}$$

where λ is the balanced weight. The conventional probability is computed as the original Transformer does, basing on sublayers of feed-forward, linear projection, and softmax stacked over the conventional attention as illustrated in the left part of Figure 1. The opponent probability is based on similar sublayers stacked over the opponent attention, but with softmin as the last sublayer as illustrated in the right part of Figure 1.

Due to the contrary properties of softmax and softmin, jointly maximizing P_c and P_o actually maximizes the contribution from the conventional attention for summary word generation, while at the same time minimizes the contribution from the opponent attention[3]. In other words, the training objective is to let the relevant part attended by $\mathrm{attention}_c$ contribute more to the summarization, while let the irrelevant or less relevant parts attended by $\mathrm{attention}_o$ contribute less.

During decoding, we aim to find maximum $\mathcal{J}$ of Equation (9) in the beam search process.

4 Experiments

We conduct experiments on abstractive sentence summarization benchmark datasets to demonstrate the effectiveness of the proposed contrastive attention mechanism.

4.1 Datasets

In this paper, we evaluate our proposed method on three abstractive text summarization benchmark datasets. First, we use the annotated Gigaword corpus and preprocess it identically to Rush *et al.* (2015), which results in around 3.8M training samples, 190K validation samples and 1951 test samples for evaluation. The source-summary pairs are formed through pairing the first sentence of each article with its headline. We use DUC-2004 as another English data set only for testing in our experiments. It contains 500 documents, each containing four human-generated reference summaries. The length of the summary is capped at 75 bytes. The last data set we used is a large corpus of Chinese short text summarization (LCSTS) (Hu et al., 2015), which is collected from the Chinese microblogging website Sina Weibo. We follow the data split of the original paper, with 2.4M source-summary pairs from the first part of the corpus for training, 725 pairs from the last part with high annotation score for testing.

4.2 Experimental Setup

We employ Transformer as our basis architecture[4]. Six layers are stacked in both the encoder and decoder, and the dimensions of the embedding vectors and all hidden vectors are set 512. The inner layer of the feed-forward sublayer has the dimensionality of 2048. We set eight heads in the multi-head attention. The source embedding, the target embedding and the linear sublayer are shared in our experiments. Byte-pair encoding is employed in the English experiment with a shared source-target vocabulary of about 32k tokens (Sennrich et al., 2015).

Regarding the contrastive attention mechanism, the opponent attention is derived from the head

[3] We also tried replacing softmin in Equation (7) with softmax, and correspondingly setting the training objective as maximizing $\mathcal{J} = \log(P_c) - \lambda\log(P_o)$, but this method failed to train because P_o becomes too small during training, and results in negative infinity value of $\log(P_o)$ that hampers the training. In comparison, softmin and the training objective of Equation (9) do not have such problem, enabling the effective training of the proposed network.

[4] https://github.com/pytorch/fairseq

System	Gigaword			DUC2004		
	R-1	R-2	R-L	R-1	R-2	R-L
ABS (Rush et al., 2015)	29.55	11.32	26.42	26.55	7.06	22.05
ABS+ (Rush et al., 2015)	29.76	11.88	26.96	28.18	8.49	23.81
RAS-Elman (Chopra et al., 2016)	33.78	15.97	31.15	28.97	8.26	24.06
words-lvt5k-1sent (Nallapati et al., 2016)	35.30	16.64	32.62	28.61	9.42	25.24
SEASS$_{\text{beam}}$ (Zhou et al., 2017)	36.15	17.54	33.63	29.21	9.56	25.51
RNN$_{\text{MRT}}$ (Ayana et al., 2016)	36.54	16.59	33.44	30.41	10.87	26.79
Actor-Critic (Li et al., 2018)	36.05	17.35	33.49	29.41	9.84	25.85
StructuredLoss (Edunov et al., 2018)	36.70	17.88	34.29	-	-	-
DRGD (Li et al., 2017)	36.27	17.57	33.62	31.79	10.75	27.48
ConvS2S (Gehring et al., 2017)	35.88	17.48	33.29	30.44	10.84	26.90
ConvS2S$_{\text{ReinforceTopic}}$ (Wang et al., 2018)	36.92	18.29	34.58	31.15	10.85	**27.68**
FactAware (Cao et al., 2018)	37.27	17.65	34.24	-	-	-
Transformer	37.87	18.69	35.22	31.38	10.89	27.18
Transformer+ContrastiveAttention	**38.72**	**19.09**	**35.82**	**32.22**	**11.04**	27.59

Table 1: ROUGE scores on the English evaluation sets of both Gigaword and DUC2004. On Gigaword, the full-length F-1 based ROUGE scores are reported. On DUC2004, the recall based ROUGE scores are reported. "-" denotes no score is available in that work.

whose attention is most synchronous to word alignments of the source-summary pair. In our experiments, we select the fifth head of the third layer for deriving the opponent attention in the English experiments, and select the second head of the third layer in the Chinese experiments. All dimensions in the contrastive architecture are set 64. The λ in Equation (9) is tuned on the development set in each experiment.

During training, we use the Adam optimizer with $\beta 1$ = 0.9, $\beta 2$ = 0.98, ε= 10^{-9}. The initial learning rate is 0.0005. The inverse square root schedule is applied for initial warm up and annealing (Vaswani et al., 2017). During training, we use a dropout rate of 0.3 on all datasets.

During evaluation, we employ ROUGE (Lin, 2004) as our evaluation metric. Since standard Rouge package is used to evaluate the English summarization systems, we also follow the method of Hu *et al.* (2015) to map Chinese words into numerical IDs in order to evaluate the performance on the Chinese data set.

4.3 Results

4.3.1 English Results

The experimental results on the English evaluation sets are listed in Table 1. We report the full-length F-1 scores of ROUGE-1 (R-1), ROUGE2 (R-2), and ROUGE-L (R-L) on the evaluation set of the annotated Gigaword, while report the recall-based scores of the R-1, R-2, and R-L on the evaluation set of DUC2004 to follow the setting of the previous works.

The results of our works are shown at the bottom of Table 1. The performances of the related works are reported in the upper part of Table 1 for comparison. ABS and ABS+ are the pioneer works of using neural models for abstractive text summarization. RAS-Elman extends ABS/ABS+ with attentive CNN encoder. words-lvt5k-1sent uses large vocabulary and linguistic features such as POS and NER tags. RNN$_{\text{MRT}}$, Actor-Critic, StructuredLoss are sequence-level training methods to overcome the problem of the usual teacher-forcing methods. DRGD uses recurrent latent random model to improve summarization quality. FactAware generates summary words conditioned on both the source text and the fact descriptions extracted from OpenIE or dependencies. Besides the above RNN-based related works, CNN-based architectures of ConvS2S and ConvS2S$_{\text{ReinforceTopic}}$ are included for comparison.

Table 1 shows that we build a strong baseline using Transformer alone which obtains the state-of-the-art performance on Gigaword evaluation set, and obtains comparable performance to the state-of-the-art on DUC2004. When we introduce the contrastive attention mechanism into Transformer, it significantly improves the performance of Transformer, and greatly advances the state-of-the-art on both Gigaword evaluation set and DUC2004, as shown in the row of "Transformer+Contrastive Attention".

4.3.2 Chinese Results

Table 2 presents the evaluation results on LCSTS. The upper rows list the performances of the related works, the bottom rows list the perfor-

System	R-1	R-2	R-L
RNN context (Hu et al., 2015)	29.90	17.40	27.20
CopyNet (Gu et al., 2016)	34.40	21.60	31.30
RNN$_{\text{MRT}}$ (Ayana et al., 2016)	38.20	25.20	35.40
RNN$_{\text{distraction}}$ (Chen et al., 2016)	35.20	22.60	32.50
DRGD (Li et al., 2017)	36.99	24.15	34.21
Actor-Critic (Li et al., 2018)	37.51	24.68	35.02
Global (Lin et al., 2018)	39.40	26.90	36.50
Transformer	41.93	28.28	38.32
Transformer+ContrastiveAttention	**44.35**	**30.65**	**40.58**

Table 2: The full-length F-1 based ROUGE scores on the Chinese evaluation set of LCSTS.

mances of our Transformer baseline and the integration of the contrastive attention mechanism into Transformer. We only take character sequences as source-summary pairs and evaluate the performance based on reference characters for strict comparison to the related works.

Table 2 shows that Transformer also sets a strong baseline on LCSTS that surpasses the performances of the previous works. When Transformer is equipped with our proposed contrastive attention mechanism, the performance is significantly improved and drastically advances the state-of-the-art on LCSTS.

5 Analysis and Discussion

5.1 Effect of the Contrastive Attention Mechanism on Attentions

Figure 3 shows the attention weights before and after using the contrastive attention mechanism. We depict the averaged attention weights of all heads in one layer in Figure 3a and 3b to study how it contributes to the conventional probability computation, and depict the opponent attention weights in Figure 3c to study its contribution to the opponent probability. Since we select the fifth head of the third layer to derive the opponent attention in English experiment, the studies are carried out on the third layer.

Figure 3a is from the baseline Transformer, Figure 3b is from "Transformer + ContrastiveAttention". We can see that "Transformer + ContrastiveAttention" is more focused on the source parts that are most relevant to the summary than the baseline Transformer, which scatters attention weights on summary word neighbors or even functional words such as "-lrb-" and "the". "Transformer + ContrastiveAttention" cancels such scattered attentions by using the contrastive attention mechanism.

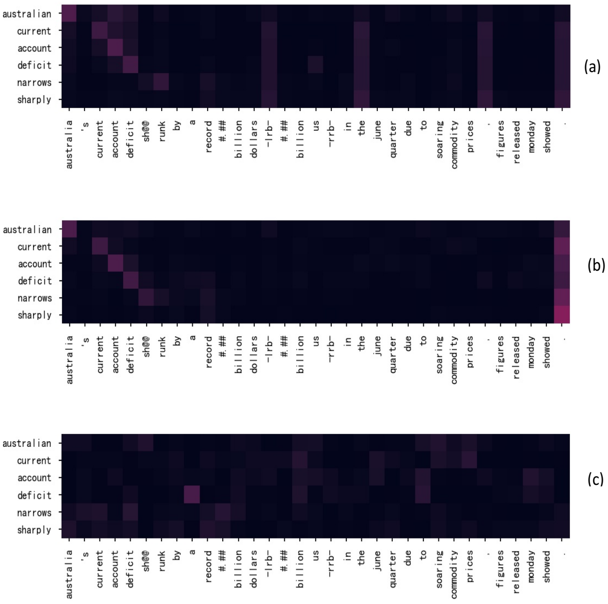

Figure 3: The attention weight changes by using the contrastive attention mechanism. (a) is the average attention weights of the third layer of the baseline Transformer, (b) is that of "Transformer+ContrastiveAttention", and (c) is the opponent attention derived from the fifth head of the third layer.

Figure 3c depicts the opponent attention weights. They are optimized during training to generate the lowest score which is fed into softmin to get the highest opponent probability P_o. The more irrelevant to the summary word the opponent is, the lower the score can be obtained, thus resulting in higher P_o. Figure 3c shows that the attentions are formed over irrelevant parts with varied weights as the result of maximizing P_o during training.

5.2 Effect of the Opponent Probability in Decoding

We study the contribution of the opponent probability P_o by dropping it during decoding to see if it hurts the performance. Table 4 shows that dropping P_o significantly harms the performance of "Transformer + ContrastiveAtt". The performance difference between the model dropping P_o and the baseline Transformer is marginal, indicating that adding the opponent probability P_o is key for achieving the performance improvement.

5.3 Explorations on Deriving the Opponent Attention

Masking More Attention Weights for Deriving the Opponent Attention

System	Gigaword			DUC2004		
	R-1	R-2	R-L	R-1	R-2	R-L
mask maximum weight	38.72	19.09	35.82	32.22	11.04	27.59
mask top-2 weights	38.17	19.15	35.51	31.87	10.94	27.41
mask top-3 weights	38.36	19.11	35.56	31.67	10.37	27.31
dynamically mask	38.12	18.92	35.28	31.37	10.32	27.11
synchronous head	38.72	19.09	35.82	32.22	11.04	27.59
non-synchronous head	37.85	18.59	35.16	31.73	10.74	27.35
averaged head	38.43	19.10	35.53	31.82	10.98	27.43
Transformer baseline	37.87	18.69	35.22	31.38	10.89	27.18

Table 3: Results of explorations on the opponent attention derivation. The upper part presents the influence of masking more attention weights for deriving the opponent attention. The middle part presents the results of selecting different head for the opponent attention derivation. The bottom row presents the result of Transformer.

Gigaword	R-1	R-2	R-L
Transformer	37.87	18.69	35.22
Transformer+ContrastiveAtt-P_o	37.92	18.88	35.21
Transformer+ContrastiveAtt	38.72	19.09	35.82
DUC2004	R-1	R-2	R-L
Transformer	31.38	10.89	27.18
Transformer+ContrastiveAtt-P_o	31.21	10.70	26.85
Transformer+ContrastiveAtt	32.22	11.04	27.59

Table 4: The effect of dropping P_o (denoted by -P_o) from Transformer+ContrastiveAtt during decoding.

In Section 3.2.1, we mask the most salient word that has the maximum weight of α_c to derive the opponent attention. In this subsection, we experimented with masking more weights of α_c by two ways: 1) masking top k weights, 2) dynamically masking. In the dynamically masking method, we order the weights from big to small at first, then go on masking two neighbors until the ratio between them is over a threshold. The threshold is 1.02 based on training and tuning on the development set.

The upper rows of Table 3 presents the performance comparison between masking maximum weight and masking more weights. It shows that masking maximum weight performs better, indicating that masking the most salient weight leaves more irrelevant or less relevant words to compute the opponent probability P_o, which is more reliable than that computed from less remaining words after masking more weights.

Selecting Non-synchronous Head or Averaged Head for Deriving the Opponent Attention

As explained in Section 3.2.1, the opponent attention is derived from the head that is most synchronous to the word alignments between source sentence and summary. We denote it "synchronous head". We also explored deriving the opponent attention from the fifth head of the first layer, which is non-synchronous to the word alignments as illustrated in Figure 2b. Its result is presented in the "non-synchronous head" row. In addition, the attention weights averaged on all heads of the third layer are used to derive the opponent attention. We denote it "averaged head".

As shown in the middle part of Table 3, both "non-synchronous head" and "averaged head" underperform "synchronous head". "non-synchronous head" performs worst, and even worse than the Transformer baseline on Gigaword. This indicates that it is better to compose the opponent attention from irrelevant parts that can be easily located in the synchronous head. "averaged head" performs slightly worse than "synchronous head", and is also slower due to the involved all heads.

5.4 Qualitative Study

Table 5 shows the qualitative results. The highlights in the baseline Transformer manifest the incorrect areas extracted by the baseline system. In contrast, the highlights in Transformer+ContrastiveAtt show that correct contents are extracted since the contrastive system distinguish relevant parts from irrelevant parts on the source side and made attending to correct areas more easily.

6 Conclusion

We proposed a contrastive attention mechanism for abstractive sentence summarization, using both the conventional attention that attends to the relevant parts of the source sentence, and a novel opponent attention that attends to irrelevant or less relevant parts for the summary word generation. Both categories of the attention constitute a contrastive pair, and we encourage contribution from the conventional attention and penalize con-

Src:press freedom in algeria remains at risk despite the release on wednesday of prominent newspaper editor mohamed UNK after a two-year prison sentence , human rights organizations said .
Ref:algerian press freedom at risk despite editor 's release UNK picture
Transformer:press freedom remains at risk in algeria rights groups say
Transformer+ContrastiveAtt:press freedom remains at risk despite release of algerian editor

Src:denmark 's poul-erik hoyer completed his hat-trick of men 's singles badminton titles at the european championships , winning the final here on saturday
Ref:hoyer wins singles title
Transformer:hoyer completes hat-trick
Transformer+ContrastiveAtt:hoyer wins men 's singles title

Src:french bank credit agricole launched on tuesday a public cash offer to buy the ## percent of emporiki bank it does not already own , in a bid valuing the greek group at #.# billion euros (#.# billion dollars) .
Ref:credit agricole announces #.#-billion-euro bid for greek bank emporiki
Transformer:credit agricole launches public cash offer for greek bank
Transformer+ContrastiveAtt:french bank credit agricole bids #.# billion euros for greek bank

Table 5: Example summaries generated by the baseline Transformer and Transformer+ContrastiveAtt.

tribution from the opponent attention through joint training. Using Transformer as a strong baseline, experiments on three benchmark data sets show that the proposed contrastive attention mechanism significantly improves the performance, advancing the state-of-the-art performance for the task.

Acknowledgments

The authors would like to thank the anonymous reviewers for the helpful comments. This work was supported by National Key R&D Program of China (Grant No. 2016YFE0132100), National Natural Science Foundation of China (Grant No. 61525205, 61673289).

References

Ayana, Shiqi Shen, Yu Zhao, Zhiyuan Liu, and Maosong Sun. 2016. Neural headline generation with sentence-wise optimization. *Computer Research Repository*, arXiv:1604.01904. Version 2.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations*.

Michele Banko, Vibhu O Mittal, and Michael J Witbrock. 2000. Headline generation based on statistical translation. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 318–325.

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: Fact aware neural abstractive summarization. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Qian Chen, Xiao Dan Zhu, Zhen Hua Ling, Si Wei, and Hui Jiang. 2016. Distraction-based neural networks for modeling document. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 2754–2760.

Sumit Chopra, Michael Auli, and Alexander M Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.

Trevor Cohn and Mirella Lapata. 2008. Sentence compression beyond word deletion. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 137–144.

Bonnie Dorr, David Zajic, and Richard Schwartz. 2003. Hedge trimmer: A parse-and-trim approach to headline generation. In *Proceedings of the HLT-NAACL 03 on Text summarization workshop-Volume 5*, pages 1–8.

Sergey Edunov, Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Classical structured prediction losses for sequence to sequence learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 355–364.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1243–1252.

Jiatao Gu, Zhengdong Lu, Li Hang, and Victor O. K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. Lcsts: A large scale chinese short text summarization dataset. In *Proceedings of the 2015 Conference on*

Empirical Methods in Natural Language Processing, pages 1967–1972.

Hongyan Jing and Kathleen R McKeown. 2000. Cut and paste based text summarization. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 178–185.

Kevin Knight and Daniel Marcu. 2000. Statistics-based summarization - step one: Sentence compression. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on on Innovative Applications of Artificial Intelligence*, pages 703–710.

Piji Li, Lidong Bing, and Wai Lam. 2018. Actor-critic based training framework for abstractive summarization. *Computing Research Repository*, arXiv:1803.11070.

Piji Li, Wai Lam, Lidong Bing, and Zihao Wang. 2017. Deep recurrent generative decoder for abstractive text summarization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2091–2100.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proc of the ACL-04 Workshop on Text Summarization Branches Out*.

Junyang Lin, Sun Xu, Shuming Ma, and Su Qi. 2018. Global encoding for abstractive summarization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 163–169.

Ramesh Nallapati, Bowen Zhou, Cicero Nogueira Dos Santos, Caglar Gulcehre, and Xiang Bing. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290.

Joel Larocca Neto, Alex A Freitas, and Celso A. A. Kaestner. 2002. Automatic text summarization using a machine learning approach. In *Brazilian Symposium on Artificial Intelligence*, pages 205–215.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. 2018a. Mask-guided contrastive attention model for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1179–1188.

Kaiqiang Song, Lin Zhao, and Fei Liu. 2018b. Structure-infused copy mechanisms for abstractive summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1717–1729.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Li Wang, Junlin Yao, Yunzhe Tao, Li Zhong, Wei Liu, and Qiang Du. 2018. A reinforced topic-aware convolutional sequence-to-sequence model for abstractive text summarization. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence,*, pages 4453–4460.

Kristian Woodsend, Yansong Feng, and Mirella Lapata. 2010. Title generation with quasi-synchronous grammar. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 513–523.

Qingyu Zhou, Yang Nan, Furu Wei, and Zhou Ming. 2017. Selective encoding for abstractive sentence summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1095–1104.